UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/608,869	06/27/2003	Kirt Debique	302128.01	7494

22971 7590 04/01/2008

MICROSOFT CORPORATION
ONE MICROSOFT WAY
REDMOND, WA 98052-6399

EXAMINER	
INGVOLDSTAD, BENNETT	
ART UNIT	PAPER NUMBER
2623	

NOTIFICATION DATE	DELIVERY MODE
04/01/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

roks@microsoft.com
ntovar@microsoft.com
a-rydore@microsoft.com

Office Action Summary	Application No.	Applicant(s)	
	10/608,869	DEBIQUE ET AL.	
	Examiner	Art Unit	
	BENNETT INGVOLDSTAD	2623	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-46* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-46* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *27 June 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *9/15/003,5/31/2005*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Claim Objections

1. Claims 18-22 and 32 are objected to because of the following informalities:

Claims 18-22: "the core layer" lacks antecedent basis.

Claim 32: "one or media sink components" should be corrected to --one or more media sink components--.

2. Appropriate correction is required.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-3, 6-8, 18, 19, 21, 23-35, 28-30, 32, 36, 37, 41-43, and 45 are rejected under 35 U.S.C. 102(e) as being anticipated by Jones et al., US 6453355.

Regarding claim 1, Jones discloses a method for processing media data, the method comprising:

- receiving one or more media data streams via a control layer (receiving media streams via a media handler [col. 5, l. 23-27]);
- modifying the data streams in one or more stream sinks (reassembling the packetized media data stream [col. 1, l. 9-13]);
- implementing in a media sink (a client display [col. 1, l. 29-31]) one or more state machines (time-based media [col. 1, l. 29-31] is played by a state machine, states including play, stop, etc.) to control a state of transfer of the media data streams (requesting the data [col. 5, l. 23-27]), the media sink responding to the control layer (the media handler control layer interprets the data [col. 5, l. 23-27]); and
- using the state of the media data streams to modify the functionality of the stream sinks (when the streaming data has reached an ending state, the client stops playing the streamed data [col. 8, l. 49-51]).

Regarding claim 2, depending on claim 1, Jones further discloses wherein the modification of the data streams is dynamic (the streaming data is played as it is received [col. 8, l. 42-51], i.e. it is depacketized dynamically).

Regarding claim 3, depending on claim 1, Jones further discloses:

- throttling the processing via the stream sinks based on one or more media sink components (time-based video [col. 1, l. 29-31] processing is throttled so that video is presented at a proper frame rate and refresh rate for the display).

Regarding claim 6, depending on claim 1, Jones further discloses wherein the media sink directs multiplexing of two or more of the media data streams into a same media sink (media data may be in a multiplexed display format [col. 12, l. 37-38]).

Regarding claim 7, depending on claim 1, Jones further discloses wherein the control layer directs control and timing for the media sink and the stream sinks (media handler directs control and timing [col. 5, l. 23-42]).

Regarding claim 8, depending on claim 1, Jones further discloses wherein the control layer directs format negotiation to be performed in the stream sinks, the format appropriate for an output device (appropriate media handler control layers access media based on a format of audio or video [col. 5, l. 23-30] and direct the streams to appropriate output devices for display [col. 1, l. 29-31]).

Regarding claim 18, depending on claim 1, Jones further discloses wherein the core layer (Quicktime media layer [col. 1, l. 29-31]) is configured to communicate

to retrieve characteristics of a sample allocator (packetized data samples may be stored at the receiving client [col. 8, l. 42-49], so characteristics such as available buffer space are retrieved from the sample storage allocator).

Regarding claim 19, depending on claim 1, Jones further discloses wherein the core layer is configured to request that a sample allocator acquire any needed resources (acquiring memory for storage [col. 8, l. 42-49]).

Regarding claim 21, depending on claim 1, Jones further discloses wherein the core layer is configured to request that a sample allocator retrieve one or more of a maximum number of samples in a sample allocation and any requested samples (the stored samples are reassembled [col. 8, l. 58-60], so the sample allocator retrieves any requested samples).

Regarding claim 23, Jones discloses a computer readable medium having computer-executable instructions [col. 7, l. 47-49] for processing data through a collection of one or more media objects, the computer-executable instructions performing acts comprising:

- receiving one or more media data streams via a control layer (receiving media streams via a media handler [col. 5, l. 23-27]);
- modifying the data streams in one or more stream sinks (reassembling the packetized media data streams [col. 1, l. 9-13]);

- implementing in a media sink (a client display [col. 1, l. 29-31]) one or more state machines (time-based media [col. 1, l. 29-31] is played by a state machine, states including play, stop, etc.) to control a state of transfer of the media data streams (requesting the data [col. 5, l. 23-27]), the media sink responding to the control layer (the media handler control layer interprets the data [col. 5, l. 23-27]); and
- using the state of the media data streams to modify the functionality of the stream sinks (when the streaming data has reached an ending state, the client stops playing the streamed data [col. 8, l. 49-51]).

Regarding claim 24, depending on claim 23, Jones further discloses wherein the modification of the data streams is dynamic (the streaming data is played as it is received [col. 8, l. 42-51], i.e. it is depacketized dynamically).

Regarding claim 25, depending on claim 23, Jones further discloses:

- throttling the processing via the stream sinks based on one or more media sink components (time-based video [col. 1, l. 29-31] processing is throttled so that video is presented at a proper frame rate and refresh rate for the display).

Regarding claim 28, depending on claim 23, Jones further discloses wherein the media sink directs multiplexing of two or more of the media data streams into a

same media sink (media data may be in a multiplexed display format [col. 12, l. 37-38]).

Regarding claim 29, depending on claim 23, Jones further discloses wherein the control layer directs control and timing for the media sink and the stream sinks (media handler directs control and timing [col. 5, l. 23-42]).

Regarding claim 30, depending on claim 23, Jones further discloses wherein the control layer directs format negotiation to be performed in the stream sinks, the format appropriate for an output device (appropriate media handler control layers access media based on a format of audio or video [col. 5, l. 23-30] and direct the streams to appropriate output devices for display [col. 1, l. 29-31]).

Regarding claim 32, Jones discloses a multimedia system comprising:

- a control layer configured to receive one or more media data streams from an application (a media handler receving media data streams [col. 5, l. 23-27] from a server [Fig 8]);
- a core layer coupled to the control layer (Quicktime media layer [col. 1, l. 29-31]), the core layer including:
    - one or media sink components configured to implement one or more state machines to control transfer of the media data streams through the multimedia system (media layer manages the display of

time-based media [col. 1, l. 29-31], which implements states including play, stop, etc.); and

- one or more stream sinks (media data stream reassemblers [col. 1, l. 9-13]) configured to dynamically modify the media data streams via the control layer (media handler interprets the data stream [col. 5, l. 23-27]) and an identified state of the media data streams determined in the media sink components (e.g., an ending state of the presentation [col. 8, l. 49-51]).

Regarding claim 36, depending on claim 32, Jones further discloses wherein the core layer is configured to communicate with the media sink to retrieve the characteristics of the media sink (the core media layer manages [col. 1, l. 20-3] the media handler by the media handler characteristics in order to use the appropriate media handler [col. 5, l. 23-27]).

Regarding claim 37, depending on claim 32, Jones further discloses wherein the core layer is configured to communicate with the media sink to add an additional stream sink and remove one of the stream sinks (the media layer may manage video handlers, audio handlers, or both [col. 7, l. 33-39]).

Regarding claim 41, depending on claim 32, Jones further discloses wherein the core layer is configured to communicate to set a rate of a presentation clock and

retrieve a presentation clock setting (core layer manages the timing of the presentation [col. 40-42]).

Regarding claim 42, depending on claim 32, Jones further discloses wherein the core layer (Quicktime media layer [col. 1, l. 29-31]) is configured to communicate to retrieve characteristics of a sample allocator (packetized data samples may be stored at the receiving client [col. 8, l. 42-49], so characteristics such as available buffer space are retrieved from the sample storage allocator).

Regarding claim 43, depending on claim 32, Jones further discloses wherein the core layer is configured to request that a sample allocator acquire any needed resources (acquiring memory for storage [col. 8, l. 42-49]).

Regarding claim 45, depending on claim 32, Jones further discloses wherein the core layer is configured to request that a sample allocator retrieve one or more of a maximum number of samples in a sample allocation and any requested samples (the stored samples are reassembled [col. 8, l. 58-60], so the sample allocator retrieves any requested samples).

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 4, 11-17, 20, 22, 26, 33, 38-40, 44, and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jones et al., US 6453355, in view of "Quicktime 6 API Reference".

Regarding claim 4, depending on claim 1, Jones does not further specifically disclose wherein one or more of the media sink and the stream sinks provide notifications for events to the control layer.

The "Quicktime 6 API Reference" discloses that the Quicktime system stream sink automatically provides notification for packet loss events to the control layer (see Discussion section under RPTRssmSendPacketList call).

It would have been obvious to have modified the stream sink disclosed by Jones with the teaching of the stream sink as disclosed in the "Quicktime 6 API Reference" for the purpose of informing the control layer of packet loss events.

Regarding claim 11, depending on claim 1, Jones does not further disclose wherein the stream sink accesses an application programming interfaces (API) that enables the stream sink to access a pointer to the media sink.

"Quicktime 6 API Reference" discloses an API call that enables an application or component to access a pointer to a media sink (see OpenComponent call and QTVideoOutputBaseSetEchoPort call).

It would have been obvious to have modified the stream sink to have accessed a pointer to a media sink via the Quicktime API for the purpose of interacting with the Quicktime framework to control the media stream.

Claim 12, depending on claim 1, is rejected under the same grounds as claim 11 wherein the "Quicktime 6 API Reference" further discloses an application programming interfaces (API) that provides an identifier for the media sink (see the "vo" variable in the QTVideoOutputBaseSetEchoPort call).

Claim 13, depending on claim 1, is rejected under the same grounds as claim 11 wherein the "Quicktime 6 API Reference" further discloses an application programming interfaces (API) that provides a type of media in use (see Discussion section in SGGetChannelDeviceList call).

Claim 14, depending on claim 1, is rejected under the same grounds as claim 11 wherein the "Quicktime 6 API Reference" further discloses an application programming interfaces (API) configured to cause processing of a sample of the media data (see DecompressSequenceBeginS call).

Claim 15, depending on claim 1, is rejected under the same grounds as claim 11 wherein the "Quicktime 6 API Reference" further discloses an application programming interfaces (API) configured to remove any data that has not been processed (see DataHFlushCache call).

Claim 16, depending on claim 1, is rejected under the same grounds as claim 11 wherein the "Quicktime 6 API Reference" further discloses an application programming interfaces (API) configured to place a marker in the data stream to determine when the stream sink has finished processing received data associated with the marker (see "tune" variable in TuneQueue call).

Claim 17, depending on claim 1, is rejected under the same grounds as claim 11 wherein the "Quicktime 6 API Reference" further discloses an application programming interfaces (API) configured to identify an end of a segment of the media data (see "tune" variable in TuneQueue call).

Regarding claim 20, depending on claim 1, Jones does not further specifically disclose wherein the core layer is configured to request that a sample allocator end an asynchronous resource allocation process.

The "Quicktime 6 API Reference" discloses that the Quicktime core layer is configured to request that a sample allocator end an asynchronous resource allocation process (see VDReleaseAsyncBuffers call).

It would have been obvious to have combined the Quicktime core layer disclosed by Jones with the teachings of the "Quicktime 6 API Reference" for the purpose of controlling the Quicktime system media stream through the API.

Claim 22, depending on claim 1, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference" discloses that the Quicktime core layer is configured to request that a sample allocator cancel one or more allocations (see VDReleaseAsyncBuffers call).

Claim 26, depending on claim 23, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference" discloses that the Quicktime system stream sink automatically provides notification for packet loss events to the control layer (see Discussion section under RPTRssmSendPacketList call).

Claim 33, depending on claim 32, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference" discloses that the control layer disclosed by Jones is an application programming interface (API) (See media handler API calls, e.g. GetMediaHandler call).

Claim 38, depending on claim 32, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference"

discloses that the core layer disclosed by Jones is configured to communicate with the media sink, the media sink enabled to report the number of stream sinks associated with a given media sink (see QTVideoOutputGetCurrentClientName call).

Claim 39, depending on claim 32, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference" discloses that the core layer disclosed by Jones is configured to communicate with the stream sinks to send a pointer to a stream sink associated with the media sink by an index in the media sink (See SGGetChannelDeviceList call, where the sequence grabber channel is associated with the stream sink).

Claim 40, depending on claim 32, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference" discloses that the core layer disclosed by Jones is configured to communicate to send a pointer to a stream sink associated with the media sink using a stream sink identifier (See SGGetChannelDeviceList call, where a stream sink identifier "c" is provided).

Claim 44, depending on claim 32, is rejected continuing with the grounds as set forth for claim 20, wherein Jones in view of the "Quicktime 6 API Reference" discloses that the Quicktime core layer disclosed by Jones is configured to

request that a sample allocator end an asynchronous resource allocation process (see VDReleaseAsyncBuffers call).

Claim 46, depending on claim 1, is rejected continuing with the grounds as set forth for claim 20 "Quicktime 6 API Reference" discloses that the Quicktime core layer disclosed by Jones is configured to request that a sample allocator cancel one or more allocations (see VDReleaseAsyncBuffers call).

7. Claims 5 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jones et al., US 6453355, in view of Dahley et al., US 2005/0132408.

Regarding claims 5 and 27, depending respectively on claims 1 and 23, Jones does not specifically disclose wherein the media data stream switches to a second media sink upon a detection of invalid media sink.

Dahley discloses a hardware multimedia framework analogous to the software multimedia framework disclosed by Jones. Dahley's system switches between output data sinks based on whether the data sink is detected as valid or invalid. See para [0107].

It would have been obvious to have implemented the output-switching technique in the system of Jones for the purpose of enabling the system to recognize the connection and disconnection of the media sinks and respond appropriately.

8. Claims 9-10, 31, and 34-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jones et al., US 6453355, in view of "An Introduction to Quicktime".

Regarding claims 9, 31, and 34, depending respectively on claims 1, 23, and 32, Jones further discloses wherein the control layer includes a media engine and a media processor (media handler accesses and interprets media [col. 5, l. 23-30]), the media engine communicating with a core layer (with the core Quicktime media layer [col. 1, l. 29-31]) to direct a pipeline to the media sink (for display [col. 1, l. 29-31]).

Jones does not further disclose directing a pipeline through one or more multimedia transforms.

"An Introduction to Quicktime" discloses that the Quicktime system provides multimedia transforms for operation on a pipeline (realtime Filter Effects [Effects webpage]).

It would have been obvious to have modified the media engine to have directed a pipeline through the multimedia transform as taught by the secondary reference for the purpose of applying filter effects to the multimedia stream.

Regarding claims 10 and 35, depending respectively on claims 9 and 32, Jones in view of "An Introduction to Quicktime" further discloses wherein the core layer (Quicktime media layer [col. 1, l. 29-31]) includes the media sink (e.g. display

[col. 1, l. 29-31]), stream sinks (management [col. 1, l. 29-31] of received data [col. 5, l. 23-30]), a media source (media data source 764 [Fig 13], managed by Quicktime [col. 1, l. 29-31]), the multimedia transforms (effects filter [Effects webpage]) and stream sources (streaming protocol handlers [col. 5, l. 12-16]).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to BENNETT INGVOLDSTAD whose telephone number is (571)270-3431. The examiner can normally be reached on M-Th 8-6:30 EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Scott Beliveau can be reached on (571) 272-7343. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

BI

/Scott Beliveau/
Supervisory Patent Examiner, Art Unit 2623